

La Biblioteca di Atlantide

Quaderni di letteratura grigia

(CC) 2005. Quest'opera è stata rilasciata sotto la licenza
Creative Commons Attribuzione - Non commerciale - Non opere derivate.
Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/publicdomain/>
o spedisce una lettera a
Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

www.sergiofumich.com

Ca' "La Gatera"

26822 Brembio (LO) – Via Togliatti, 3

Sergio Fumich

NUMERAZIONE E CODICI

APPUNTI DI MATEMATICA
APPLICATA AL CALCOLATORE

Ca' "La Gatera"

2005

SISTEMI DI NUMERAZIONE

Un primo modo per rappresentare un numero naturale n diverso da zero può essere quello di usare una famiglia di elementi, ciascuno indicato con $|$ il cui cardinale sia lo stesso numero n . Invece di elementi si può parlare di *unità*.

Tale sistema di rappresentazione tuttavia è inadeguato per i nostri sensi e, dunque, siamo spinti a studiare rappresentazioni abbreviate dei numeri naturali, le quali costituiscono la teoria della numerazione. Semplici considerazioni suggeriscono che la scelta per la base b del sistema di numerazione di un numero sensibilmente diverso dal numero dieci risulterebbe sconveniente per l'uomo. Se, infatti, la base b fosse molto minore di dieci, i numeri usati nella vita quotidiana raggiungerebbero una lunghezza eccessiva per la memoria umana. D'altra parte, se la base fosse sensibilmente maggiore di dieci, la tavola pitagorica diventerebbe un peso eccessivo per la memoria.

Il problema fondamentale della numerazione può essere espresso nel modo seguente:

data una famiglia ordinata e finita di b simboli distinti ($U_0, U_1, U_2, \dots, U_{b-1}$) che raffigurano rispettivamente i numeri $0, 1, 2, \dots, b-1$, rappresentare nel modo più semplice possibile ogni numero naturale n mediante questi simboli.

Gli elementi U_i si chiamano *cifre* o *figure* e b si chiama la *base del sistema di numerazione*.

Una soluzione è quella accennata in precedenza, dove si usa il simbolo $|$ per rappresentare i numeri naturali diversi da zero, ma abbiamo detto che tale rappresentazione non è, per noi, molto conveniente. Supporremo, dunque, che sia $\mathbf{b} > 1$.

La famiglia delle cifre ($\mathbf{U}_0, \mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_{\mathbf{b}-1}$) e i numeri naturali dell'intervallo $[0, \mathbf{b}-1]$ sono in corrispondenza biunivoca. Si rappresenta, dunque, ogni numero naturale $\mathbf{n} < \mathbf{b}$ con la cifra che corrisponde ad $\mathbf{n}$. Per tutti gli altri numeri naturali, la rappresentazione consiste nel fare corrispondere a ciascun numero naturale una successione di cifre.

Si dimostra il seguente teorema fondamentale:

ogni numero naturale $\mathbf{n}$ può scriversi nella forma:

$$\mathbf{n} = \mathbf{a}_m \mathbf{b}^m + \mathbf{a}_{m-1} \mathbf{b}^{m-1} + \dots + \mathbf{a}_1 \mathbf{b} + \mathbf{a}_0$$

dove è $\mathbf{m} \geq 0$, $0 \leq \mathbf{a}_k \leq \mathbf{b}-1$ ($\mathbf{k} = 0, 1, \dots, \mathbf{m}$), ed è $\mathbf{a}_m > 0$ se è $\mathbf{m} > 0$. Tale rappresentazione è unica.

Il teorema garantisce che esiste un unico $\mathbf{m}$ ed un'unica $(\mathbf{m}+1)$ -pla $(\mathbf{a}_m, \mathbf{a}_{m-1}, \dots, \mathbf{a}_1, \mathbf{a}_0)$ per cui è vera l'uguaglianza di sopra. Pertanto possiamo indicare il numero $\mathbf{n}$ con la $(\mathbf{m}+1)$ -pla; convenzionalmente sarà:

$$\mathbf{n} = (\mathbf{a}_m, \mathbf{a}_{m-1}, \dots, \mathbf{a}_1, \mathbf{a}_0)_b$$

e poiché è $0 \leq \mathbf{a}_i \leq \mathbf{b}-1$ possiamo, tenendo conto della corrispondenza biunivoca tra la famiglia di cifre e l'intervallo $[0, \mathbf{b}-1]$, rappresentare il numero con una $(\mathbf{m}+1)$ -pla di cifre $\mathbf{U}$.

In generale, passando dall'insieme $\mathbf{N}$ all'insieme $\mathbf{Q}$ dei numeri razionali, detta $\mathbf{b}$ la base del sistema di numerazione, il numero

$$\mathbf{a}_k \mathbf{a}_{k-1} \dots \mathbf{a}_0, \mathbf{a}_{-1} \mathbf{a}_{-2} \dots \mathbf{a}_{-h}$$

in base $\mathbf{b}$ rappresenterà il valore:

$$\mathbf{a}_k \mathbf{b}^k + \mathbf{a}_{k-1} \mathbf{b}^{k-1} + \dots + \mathbf{a}_0 \mathbf{b}^0 + \mathbf{a}_{-1} \mathbf{b}^{-1} + \dots + \mathbf{a}_{-h} \mathbf{b}^{-h}$$

La numerazione in base $\mathbf{2}$ (o *binaria*) utilizza i due *bit*¹ $\mathbf{0}$ e $\mathbf{1}$. Il passaggio da un numero espresso in base $\mathbf{2}$ al corrispondente in base $\mathbf{10}$ è immediato:

$$(\mathbf{1001})_2 = \mathbf{1} \cdot \mathbf{2}^3 + \mathbf{0} \cdot \mathbf{2}^2 + \mathbf{0} \cdot \mathbf{2}^1 + \mathbf{1} \cdot \mathbf{2}^0 = (\mathbf{9})_{10}$$

Si può utilizzare anche il seguente schema di calcolo derivato dallo schema di Horner per il calcolo del valore di un polinomio:

	1	0	0	1	
2	2	4	8		
	1	2	4	9	← risultato della trasformazione

La conversione di un numero decimale intero nel suo equivalente binario può essere fatta con il seguente algoritmo:

```

begin
    leggi numero
    i = 0
    do

```

¹ Bit è la contrazione di *binary digit*, cifra binaria.

```

    if resto(numero/2) = 0
        then bit-di-peso(i) = 0
        else bit-di-peso(i) = 1
    endif
    numero = parte-intera(numero/2)
    i = i + 1
repeat until numero = 0
end

```

L'algoritmo costruisce il numero binario a partire dalla cifra meno significativa.

Esempio:

	273	resti
	136	1 = a_0 (bit meno significativo)
	68	0 = a_1
	34	0 = a_2
quozienti	17	0 = a_3
	8	1 = a_4
	4	0 = a_5
	2	0 = a_6
	1	0 = a_7
	0	1 = a_8 (bit più significativo)

$$(273)_{10} = (100010001)_2$$

Per quanto riguarda la conversione di numeri frazionari, la conversione da binario a decimale non necessita di particolari accorgimenti:

$$(101,001)_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3}$$

cioè: $(101,001)_2 = (5,125)_{10}$

La trasformazione inversa può essere fatta convertendo separatamente la parte intera e la parte frazionaria. Per la prima conversione si applica il precedente algoritmo, per la seconda l'algoritmo seguente:

begin

leggi numero

i = 1

do

if numero * 2 ≥ 1

then bit-di-peso(i) = 1

else bit-di-peso(i) = 0

endif

numero = parte-frazionaria(numero * 2)

i = i + 1

repeat until numero = 0

end

L'algoritmo costruisce la parte frazionaria a partire dal bit più significativo.

Esempio:

$$(17,125)_{10} \rightarrow (?)_2$$

Effettuiamo la conversione della parte intera:

17		
8	1	Bit meno significativo
4	0	
2	0	
1	0	
0	1	Bit più significativo

$$(17)_{10} = (10001)_2$$

Effettuiamo ora la conversione della parte frazionaria:

125		
250	0	Bit più significativo
500	0	
000	1	Bit meno significativo

$$(0,125)_{10} = (0,001)_2$$

Pertanto:

$$(17,125)_{10} = (10001,001)_2$$

Le basi **8** e **16** vengono introdotte perché la rappresentazione binaria dei numeri, per quanto adatta al calcolatore, non è altrettanto comoda per l'uomo che può facilmente commettere errori nel leggere o scrivere lunghe sequenze di bit.

Base 10	Base 2	Base 8	Base 16
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14

La conversione binario-esadecimale si attua dividendo i numeri binari, da destra verso sinistra, in campi di 4 bit che vengono separatamente convertiti in cifre esadecimali.

Ad esempio, il numero $(100111)_2$ si converte così:

$$10 \mid 0111 = (27)_{16}$$

La conversione dalla base **16** alla base **2** si effettua sostituendo ogni cifra esadecimale con la corrispondente configurazione binaria di 4 bit.

Esempio:

$$(92A)_{16} = 1001 \mid 0010 \mid 1010 = (100100101010)_2$$

La conversione esadecimale-decimale si effettua col solito procedimento:

$$(27)_{16} = 2 \cdot 16^1 + 7 \cdot 16^0 = (39)_{10}$$

Per passare invece dalla base **10** alla base **16** si usano algoritmi simili a quelli del caso binario.

OPERAZIONI SUI NUMERI BINARI

Tabella della addizione:

+	0	1
0	0	1
1	1	(1)0

Esempio:

$$\begin{array}{r}
 \text{riporti} \quad 111 \\
 \text{a} \quad 110101 \quad + \\
 \text{b} \quad 100011 \quad = \\
 \hline
 \text{a+b} \quad 101100 \\
 \quad \quad \quad 0
 \end{array}$$

Tabella della moltiplicazione:

°	0	1
0	0	0
1	0	1

Esempio:

$$\begin{array}{r}
 a \quad 1011 \quad * \\
 b \quad 0110 \quad = \\
 \hline
 0000 \\
 1011 \\
 1011 \\
 0000 \\
 \hline
 a \cdot b \quad 1000010
 \end{array}$$

Per quanto riguarda la sottrazione, si può procedere come nel caso decimale:

$$\begin{array}{r}
 \text{presti-} \quad 1 \\
 \text{ti} \\
 a \quad 1100 \quad - \\
 b \quad 0010 \quad = \\
 \hline
 a - b \quad 1010
 \end{array}$$

Usualmente però la sottrazione viene calcolata con altri metodi, ad esempio rappresentando il sottraendo come numero negativo ed effettuando successivamente la somma.

I numeri negativi possono essere rappresentati come complemento a 2 del corrispondente numero positivo.

Il metodo più facile per ottenere il complemento a 2 di un numero binario è quello di fare prima il complemento a 1 e poi sommare a questo complemento l'unità.

Esempio:

110001110010	numero
001110001101	complemento a 1
+ 1	
001110001110	complemento a 2

Esempio di sottrazione usando il complemento a 2:

$$\begin{aligned}
 (27)_{10} - (15)_{10} &= (011011)_2 - (001111)_2 = \\
 &= (011011)_2 + (110001)_2 = \\
 &= (001100)_2 = \\
 &= (12)_{10}
 \end{aligned}$$

Poiché dei numeri rappresentabili la prima metà viene utilizzata per i positivi e la seconda metà per i negativi, il bit più significativo vale 0 per i numeri positivi e nulli, 1 per i negativi e, in tal senso, risulta un indicatore di segno.

In parole di n bit sono rappresentabili numeri di valore compreso fra $2^{n-1} - 1$ e -2^{n-1} .

Utilizzando la rappresentazione in complemento, tutte le operazioni aritmetiche possono essere eseguite purché si abbia a disposizione un dispositivo in grado di eseguire l'operazione di addizione.

CODICI

Nei calcolatori, e più in generale nei circuiti digitali, accanto alla rappresentazione binaria pura, viene usata per i numeri una rappresentazione binaria in codice delle cifre decimali.

Tipico è il codice BCD (*Binary Coded Decimal*) ottenuto traducendo in binario puro non l'intero numero preso nel suo complesso, ma le sue singole cifre decimali. In tal modo la decodificazione riguarda le singole cifre, vi sono cioè solo 10 configurazioni differenti. La codifica delle cifre è la seguente:

0	→	0000
1	→	0001
2	→	0010
3	→	0011
4	→	0100
5	→	0101
6	→	0110
7	→	0111
8	→	1000
9	→	1001

Il più elementare codice decimale è il cosiddetto *codice "1 a 10"*. In esso le dieci cifre sono subordinate a

10 bit dei quali assume valore 1 quello corrispondente alla cifra decimale da rappresentare:

0	→	0000000001
1	→	0000000010
2	→	0000000100
3	→	0000001000
4	→	0000010000
5	→	0000100000
6	→	0001000000
7	→	0010000000
8	→	0100000000
9	→	1000000000

Non è molto vantaggioso in quanto richiede un numero elevato di posizioni.

Nel codice *biquinario* sono usati solo 7 bit:

0	→	1000001	5	→	0100001
1	→	1000010	6	→	0100010
2	→	1000100	7	→	0100100
3	→	1001000	8	→	0101000
4	→	1010000	9	→	0110000

La posizione 7 viene usata per il riconoscimento degli errori e può essere tralasciata.

Il codice "2 a 5" richiede 5 posizioni. Delle $2^5=32$ configurazioni possibili vengono utilizzate soltanto quelle nelle quali compare due volte il valore 1:

0	→	11000
1	→	00011
2	→	00101
3	→	00110
4	→	01001
5	→	01010
6	→	01100
7	→	10001
8	→	10010
9	→	10100

Una categoria di codici è quella in cui la rappresentazione è ottenuta mediante un "codice ponderato" a quattro cifre binarie (*codici a tetrade*). Il codice è costituito da quattro numeri o *pesi*: $\mathbf{b_1b_2b_3b_4}$, ciascuno diverso da zero ed appartenente all'intervallo di numeri relativi $[-9,9]$. Si tratta cioè di scegliere il codice in modo che ogni cifra decimale da 0 a 9 possa essere rappresentata da una somma:

$$\mathbf{r_1b_1 + r_2b_2 + r_3b_3 + r_4b_4}$$

dove i coefficienti $\mathbf{r_i}$ sono cifre binarie 0 e 1. Il numero 0 è rappresentato evidentemente quando tutti gli $\mathbf{r_i}$ sono nulli.

Come esempio consideriamo il *codice 1125* in cui $\mathbf{b}_1 = 1$, $\mathbf{b}_2 = 1$, $\mathbf{b}_3 = 2$, $\mathbf{b}_4 = 5$. Si ha la seguente rappresentazione:

0	→	0000	
1	→	1000	oppure 0100
2	→	1100	oppure 0010
3	→	1010	oppure 0110
4	→	1110	
5	→	0001	
6	→	1001	oppure 0101
7	→	1101	oppure 0011
8	→	1011	oppure 0111
9	→	1111	

Tale codice non è univoco poiché non tutti i numeri hanno una rappresentazione unica.

Ci occuperemo ora di codici aventi tutti i pesi positivi. Supporremo, come è lecito, che:

$$0 < \mathbf{b}_1 \leq \mathbf{b}_2 \leq \mathbf{b}_3 \leq \mathbf{b}_4$$

Dimostriamo che deve essere:

$$\mathbf{b}_1 = 1$$

se così non fosse, la cifra **1** non potrebbe venir rappresentata;

$$\mathbf{b}_2 \leq \mathbf{b}_1 + 1$$

se così non fosse, cioè fosse $\mathbf{b}_2 \geq 3$, la cifra **2** non potrebbe venir rappresentata;

$$b_3 \leq b_1 + b_2 + 1$$

se così non fosse, cioè fosse $b_3 > b_1 + b_2 + 1$, la cifra $b_1 + b_2 + 1$ non potrebbe venir rappresentata;

$$b_4 \leq b_1 + b_2 + b_3 + 1$$

se così non fosse, la cifra $b_1 + b_2 + b_3 + 1$ non potrebbe venir rappresentata;

$$b_1 + b_2 + b_3 + b_4 \geq 9$$

se così non fosse, il 9 non potrebbe essere rappresentato.

Pertanto le possibili quaterne di pesi sono:

1125 1224 1234 1244 1248

1134 1225 1235 1245

1135 1226 1236 1246

1136 1233 1237 1247

Di questi codici solo l'ultimo, cioè il *codice 1248* è univoco. Tale codice è il codice BCD.

Occupiamoci ora dei codici con pesi negativi. Come primo esempio consideriamo il *codice 3567*. Conveniamo di scrivere 6 invece di -6. Abbiamo:

0	→	0000	5	→	0100
1	→	0011	6	→	0111
2	→	1110	7	→	0001
3	→	1000	8	→	1100
4	→	1011	9	→	1111

Dimostriamo che *in un codice con pesi negativi non più di due pesi possono essere negativi*. Supponiamo che $\mathbf{b}_1$, $\mathbf{b}_2$ e $\mathbf{b}_3$ siano negativi. È:

$$\mathbf{r}_1\mathbf{b}_1 + \mathbf{r}_2\mathbf{b}_2 + \mathbf{r}_3\mathbf{b}_3 < 0$$

Variando $\mathbf{r}_1$, $\mathbf{r}_2$, $\mathbf{r}_3$ in $[0,1]$, le somme possibili sono $2^3=8$, pertanto le cifre rappresentabili, che sono date da:

$$(\mathbf{r}_1\mathbf{b}_1 + \mathbf{r}_2\mathbf{b}_2 + \mathbf{r}_3\mathbf{b}_3) + \mathbf{r}_4\mathbf{b}_4$$

sono al più 9, otto casi con $\mathbf{r}_4=1$ e quello con $\mathbf{r}_1=\mathbf{r}_2=\mathbf{r}_3=\mathbf{r}_4=0$.

Dimostriamo che *condizione necessaria e sufficiente affinché il codice $\mathbf{b}_1\mathbf{b}_2\mathbf{b}_3\mathbf{b}_4$ non sia univoco è che esista una quaterna $(\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4)$ di numeri non tutti nulli scelti nell'insieme dei tre numeri $\{-1, 0, 1\}$ tali che:*

$$\mathbf{s}_1\mathbf{b}_1 + \mathbf{s}_2\mathbf{b}_2 + \mathbf{s}_3\mathbf{b}_3 + \mathbf{s}_4\mathbf{b}_4 = 0.$$

Supponiamo che il codice non sia univoco, esiste allora un numero che ha due rappresentazioni in questo codice: $\mathbf{r}_1\mathbf{r}_2\mathbf{r}_3\mathbf{r}_4$ e $\mathbf{r}'_1\mathbf{r}'_2\mathbf{r}'_3\mathbf{r}'_4$. È:

$$\mathbf{r}_1\mathbf{b}_1 + \mathbf{r}_2\mathbf{b}_2 + \mathbf{r}_3\mathbf{b}_3 + \mathbf{r}_4\mathbf{b}_4 = \mathbf{r}'_1\mathbf{b}_1 + \mathbf{r}'_2\mathbf{b}_2 + \mathbf{r}'_3\mathbf{b}_3 + \mathbf{r}'_4\mathbf{b}_4$$

da cui:

$$(\mathbf{r}_1 - \mathbf{r}'_1)\mathbf{b}_1 + (\mathbf{r}_2 - \mathbf{r}'_2)\mathbf{b}_2 + (\mathbf{r}_3 - \mathbf{r}'_3)\mathbf{b}_3 + (\mathbf{r}_4 - \mathbf{r}'_4)\mathbf{b}_4 = 0.$$

Essendo $\mathbf{r}_i$, $\mathbf{r}'_i$ appartenenti a $[0,1]$, $\mathbf{r}_i - \mathbf{r}'_i$ vale 1, 0 o -1, pertanto, ponendo $\mathbf{s}_i = \mathbf{r}_i - \mathbf{r}'_i$ si ha:

$$\mathbf{s}_1\mathbf{b}_1 + \mathbf{s}_2\mathbf{b}_2 + \mathbf{s}_3\mathbf{b}_3 + \mathbf{s}_4\mathbf{b}_4 = 0.$$

Viceversa, se esiste una quaterna $(\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4)$ che verifica la $\mathbf{s}_1\mathbf{b}_1 + \mathbf{s}_2\mathbf{b}_2 + \mathbf{s}_3\mathbf{b}_3 + \mathbf{s}_4\mathbf{b}_4 = 0$, la rappresentazione non può essere univoca. Infatti si può trovare un nu-

mero $\mathbf{a}$ avente due rappresentazioni distinte nel codice $\mathbf{b}_1\mathbf{b}_2\mathbf{b}_3\mathbf{b}_4$. Infatti, da:

$$\mathbf{a} = \mathbf{r}_1\mathbf{b}_1 + \mathbf{r}_2\mathbf{b}_2 + \mathbf{r}_3\mathbf{b}_3 + \mathbf{r}_4\mathbf{b}_4$$

e da:

$$\mathbf{0} = \mathbf{s}_1\mathbf{b}_1 + \mathbf{s}_2\mathbf{b}_2 + \mathbf{s}_3\mathbf{b}_3 + \mathbf{s}_4\mathbf{b}_4$$

discende che:

$$\mathbf{a} = (\mathbf{r}_1 + \mathbf{s}_1)\mathbf{b}_1 + (\mathbf{r}_2 + \mathbf{s}_2)\mathbf{b}_2 + (\mathbf{r}_3 + \mathbf{s}_3)\mathbf{b}_3 + (\mathbf{r}_4 + \mathbf{s}_4)\mathbf{b}_4$$

Per ipotesi tutti gli $\mathbf{s}_i$ non sono nulli. Se $\mathbf{s}_i = \mathbf{1}$, si assumerà $\mathbf{r}_i = \mathbf{0}$ e si avrà $\mathbf{r}_i + \mathbf{s}_i = \mathbf{1} \neq \mathbf{r}_i$. Se $\mathbf{s}_i = -\mathbf{1}$, si assumerà $\mathbf{r}_i = \mathbf{1}$ e si avrà $\mathbf{r}_i + \mathbf{s}_i = \mathbf{0} \neq \mathbf{r}_i$. Si avranno così ben due rappresentazioni diverse per il numero $\mathbf{a}$ così scelte.

Supponiamo che la $(\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4) = (\mathbf{1}, \mathbf{1}, -\mathbf{1}, -\mathbf{1})$ verifichi la condizione $\mathbf{s}_1\mathbf{b}_1 + \mathbf{s}_2\mathbf{b}_2 + \mathbf{s}_3\mathbf{b}_3 + \mathbf{s}_4\mathbf{b}_4 = \mathbf{0}$. È $\mathbf{b}_1 + \mathbf{b}_2 - \mathbf{b}_3 - \mathbf{b}_4 = \mathbf{0}$, cioè $\mathbf{b}_1 + \mathbf{b}_2 = \mathbf{b}_3 + \mathbf{b}_4$. Le due rappresentazioni che si ottengono sono:

$$\mathbf{0b}_1 + \mathbf{0b}_2 + \mathbf{1b}_3 + \mathbf{1b}_4 \text{ e } \mathbf{1b}_1 + \mathbf{1b}_2 + \mathbf{0b}_3 + \mathbf{0b}_4.$$

Perché il codice $\mathbf{b}_1\mathbf{b}_2\mathbf{b}_3\mathbf{b}_4$ sia univoco è necessario che, per ogni scelta degli indici $\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}$ da $\mathbf{1}$ a $\mathbf{4}$ sia:

$$|\mathbf{b}_i| \neq |\mathbf{b}_j|;$$

$$|\mathbf{b}_i| + |\mathbf{b}_j| \neq |\mathbf{b}_k|;$$

$$|\mathbf{b}_i| + |\mathbf{b}_j| + |\mathbf{b}_k| \neq |\mathbf{b}_l|.$$

Ne segue che i codici univoci si possono scegliere fra i seguenti 20 gruppi i cui pesi sono dati in valore assoluto:

1248	1249	1259	1468	1479	1579	2348	2458
2478	2489	2589	3468	3567	3678	3789	4678

4689 4789 5679 5789

Si può verificare che ci sono solo 17 codici con uno o due pesi negativi e sono:

1248 1468 2348 2458

1248 1468 2348 3468

1248 1468 2348 3567

1248 2348 2456 3567

1248

Ad esempio, il codice 1248 scritto nell'ordine 8421 è:

0	→	0000
1	→	0001
2	→	0010
3	→	0011
4	→	1100
5	→	1101
6	→	1110
7	→	1111
8	→	1000
9	→	1001

Una particolare proprietà che rende utile talvolta un certo codice è quella dell'autocomplementazione. Tale proprietà significa che se si rappresenta una cifra decimale $\mathbf{d}_i$ qualsiasi con una certa combinazione di cifre binarie del codice: $\mathbf{d}_i = \mathbf{b}_1\mathbf{b}_2\mathbf{b}_3\mathbf{b}_4$, la rappresentazio-

ne del numero $9 - d_i$ nel codice autocomplementato sarà una combinazione di bit ottenuta da quella di d_i complementando ciascun bit, cioè mettendo **0** se c'è un **1** e viceversa. Indicando convenzionalmente con **b** il complemento a **1** di **b** è: $9 - d_i = \underline{b_1} \underline{b_2} \underline{b_3} \underline{b_4}$.

Un esempio di tale codice è il codice “*eccesso-3*” che si chiama così perché si può derivarlo dal BCD sommando 3 ad ogni cifra:

0	→	0011
1	→	0100
2	→	0101
3	→	0110
4	→	0111
5	→	1000
6	→	1001
7	→	1010
8	→	1011
9	→	1100

Tale codice si chiama anche *codice Stibitz*.

Il codice 1224 è un codice pesato autocomplementante:

0	→	0000
1	→	1000
2	→	0100 oppure 0010
3	→	1100 oppure 1010

4	→	0001 oppure 0110
5	→	1110 oppure 1001
6	→	0011 oppure 0101
7	→	1011 oppure 1101
8	→	0111
9	→	1111

Tale codice solitamente è usato nella forma *2421* e reso univoco come segue:

0	→	0000
1	→	0001
2	→	0010
3	→	0011
4	→	0100
5	→	1011
6	→	1100
7	→	1101
8	→	1110
9	→	1111

Tale codice è noto come *codice Aiken*.

Un'altra proprietà che rende utile un codice decimale è quella della distanza unitaria tra due cifre decimali successive. Tale proprietà può essere utile nella individuazione e correzione di errori. Un esempio di tale codice è il *codice Gray* che appartiene anche alla categoria dei codici riflessi in cui il codice a **n** bit è gene-

rato dal codice a $n-1$ bit per riflessione, premettendo **0** alle configurazioni originali, **1** a quelle ottenute per riflessione:

0	00	000	0000
1	01	001	0001
1 bit	11	011	0011
	10	010	0010
2 bit	110	0110	
	111	0111	
	101	0101	
	100	0100	
3 bit	1100		
	1101		
	1111		
	1110		
	1010		
	1011		
	1001		
	1000		
	4 bit		

La rappresentazione delle cifre decimali con il codice Gray è la seguente:

0	→	0000
1	→	0001
2	→	0011
3	→	0010
4	→	0110
5	→	0111
6	→	0101
7	→	0100
8	→	1100
9	→	1101

Codificare una certa grandezza significa attribuire ad ogni valore, che la grandezza può assumere, una configurazione di un numero opportuno di bit, secondo una certa regola detta *codice*. Se una grandezza può assumere **M** valori distinti, il numero di bit necessario per codificare correttamente la grandezza scelta dovrà soddisfare alla relazione $2^n \geq M$.

Abbiamo visto che per codificare le cifre decimali sono necessari 4 bit, $2^3 < 10 < 2^4$. Delle possibili 16 combinazioni di cifre binarie si usano solo 10 e quindi 6 combinazioni non sono utilizzabili (parole di codice non valide). Per codificare le 26 lettere dell'alfabeto inglese e le 10 cifre numeriche sono necessarie 36 configurazioni. Essendo $2^5 < 36 < 2^6 = 64$, saranno necessari almeno 6 bit. Di solito si usano codici basati sul *byte*.

Tali sono ad esempio i codici EBCDIC (*Extended Binary Coded Interchange Code*) e ASCII (*American Standard Code for Information Interchange*).

INDICE

Sistemi di numerazione.....	5
Operazioni sui numeri binari.....	13
Codici.....	17

Notizia

Sergio Fumich è nato a Trieste nel 1947. Dal 1970 si è trasferito a Brembio, piccolo comune del Lodigiano. Lavora a Milano presso un'importante Fondazione lombarda che si occupa di formazione professionale. Ha svolto attività pubblicistica dal 1978 al 1995 come collaboratore del quotidiano di Lodi *Il Cittadino*, come direttore responsabile di alcuni fogli locali e della rivista di poesia *Keraunia*. Ha pubblicato libri di poesia e di racconti e opuscoli divulgativi.

Ca' "La Gatera"

Edizione fuori commercio

Finito di stampare a Brembio (LO) con tecniche elettroniche nel settembre 2005

Di questo opuscolo sono stati tirati 200 esemplari.